

Progress towards automated Kepler scientific workflows for computer-aided drug discovery and molecular simulations

Pek U. Ieong^{1,3}, Jesper Sørensen¹, Prasantha L. Vemu¹, Celia W. Wong¹, Özlem Demir¹, Nadya P. Williams^{2,3}, Jianwu Wang², Daniel Crawl^{2,3}, Robert V. Swift¹, Robert D. Malmstrom^{1,3}, Ilkay Altintas^{2,3} and Rommie E. Amaro^{1,3*}

¹Department of Chemistry and Biochemistry, ²San Diego Supercomputer Center, ³National Biomedical Computation Resource, University of California San Diego, CA, USA
ramaro@ucsd.edu

Abstract

We describe the development of automated workflows that support computed-aided drug discovery (CADD) and molecular dynamics (MD) simulations and are included as part of the National Biomedical Computation Resource (NBCR). The main workflow components include: file-management tasks, ligand force field parameterization, receptor-ligand molecular dynamics (MD) simulations, job submission, serial and parallel execution, and monitoring on relevant high-performance computing (HPC) resources, receptor structural clustering, virtual screening (VS), and statistical analyses of the VS results. The workflows aim to standardize simulation and analysis and promote best practices within the molecular simulation and CADD communities. Each component is developed as a stand-alone workflow, which should allow for easy integration into larger frameworks built suiting user needs, while remaining intuitive and easy to extend.

Keywords: Scientific workflows, molecular simulation, ligand parameterization, small molecule docking, structural clustering, big data reduction, web services, relaxed complex scheme

1 Introduction

Using computer simulation as an aid in drug discovery is not novel, yet the field is sometimes still considered in its infancy, an opinion that may be due to the relatively complicated processes involved

* Corresponding author: Tel: +1-858-534-9629; Fax: +1-858-534-9645; Email: ramaro@ucsd.edu

and the lack of community-wide standard procedures. Furthermore, the continuous development of new computer architectures and software parallelization can result in large amounts of data, upwards of 1 terabyte for single computer-aided drug discovery (CADD) projects. Perhaps as a result of this enabling technology, it is common for practitioners to spend more time analyzing the data than generating it. With this in mind, our aim is the development of robust, reusable workflows for simulation preparation, job execution, and analysis that simplify best practices and help the community make the most of their rich data sets.

To develop automated, standardized protocols, we employ Kepler [1], a scientific workflow framework. Kepler is a free, open-source software suite designed for analyzing and modeling scientific data. The Kepler software simplifies the creation of executable models (*scientific workflows*), even by researchers with little programming background [2]. Additionally, it is a platform for users to share and reuse data, workflows, and components for a wide range of scientific and engineering applications [3, 4]. Kepler has powerful support to handle new cyber infrastructure demands (e.g., intelligently handling/brokering access to Extreme Science and Engineering Discovery Environment (XSEDE) and other simulation-relevant platforms), and it is particularly well suited to handle workflows that cross scales. The flexibility of Kepler makes it an ideal environment for sharing methods among scientists, thus increasing reproducibility and accessibility. Kepler also provides a provenance framework (e.g. data lineage and the processing history of workflow runs) that collects information, which can be viewed through a molecular modelers' virtual notebook [5]. This latter feature also makes it possible to detail methods, software names and versions, resource specifications and computational cost in literature reports, in a straight-forward manner similar to the standardized reporting that exists for small-molecule crystal structures [6].

2 The relaxed complex scheme – main components

Previously, we developed a CADD pipeline schematic called the relaxed complex scheme (RCS), [7], an end-to-end CADD experiment that incorporates receptor flexibility into virtual screening (VS) by utilizing molecular dynamics (MD) simulations. As summarized schematically in Figure 1, the RCS facilitates all steps of VS, including: (1) generating compound libraries, (2) generating and selecting receptor structures, (3) performing virtual screens, (4) reevaluating and characterizing docked poses, and (5) sharing virtual-screening results. While not illustrated in Figure 1, workflow results lend themselves to statistical validation, an extension discussed in section 3.7

Building on our earlier RCS efforts that were designed with a specific task in-mind, we are developing individual, stand-alone workflows that are modular and reusable. Collectively, they form a “toolkit” of powerful methods that can be assembled to address a range of challenging VS problems. In particular, to incorporate protein flexibility into rational drug discovery and design, we are constructing a class of workflows to automate the setup, execution, and evaluation of molecular dynamics simulations. The workflows can be assembled in novel ways, creating environments where system-specific MD analysis can be meaningfully conducted, providing extended utility beyond CADD and the RCS. Each Kepler-based reusable workflow module is called an “actor” and is built primarily on an open-source software platform, or on software that is free to academic groups. The near universal accessibility of the workflows should translate to broad dissemination and use, allowing researchers to handle the challenges inherent in (big) data more effectively.

To prevent each workflow from becoming a “black box”, where appropriate, we are focused on including metrics or analytics that allow the user to judge the quality of the output and make key scientific decisions. As an example, we will focus on providing applications that make conducting and reporting novel MD analyses standard, routine and reproducible [8].

We furthermore plan to build workflows that support data sharing and transportation through cloud and other distributed platforms that facilitate usage of high-speed networks. The combination of these functionalities will provide a simple but powerful way to create and share customizable reports among members of large scientific collaborations.

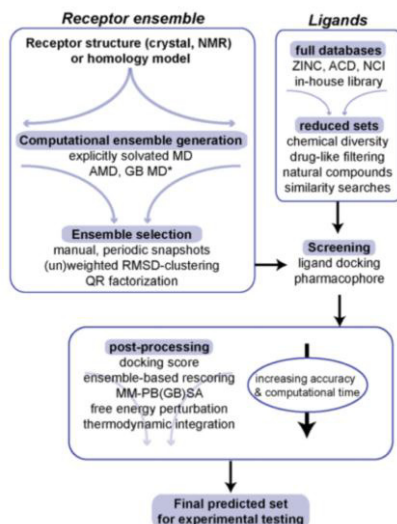


Figure 1: General workflow for ensemble-based VS experiment. Blue arrows indicate size of data sets (i.e. increasing or decreasing) at each step; * denotes emerging methods that have not yet been tested. (AMD: accelerated molecular dynamics, GB MD: generalized Born molecular dynamics, RMSD: root-mean-square-deviation, ZINC – ZINC Is Not Commercial, ACD: Available Chemical Database, NCI: National Cancer Institute, MM-PB(GB)SA: Molecular Mechanics – Poisson-Boltzmann (Generalized Born) Surface Area).

3 CADD workflow – main actors

3.1 File management for ligand parameterization

For organizational purpose, we developed a Kepler composite actor that takes a list of PDB files, which is a standardized file format containing structural information regarding molecules, and creates subdirectories using the PDB names. The PDB files are then copied to the corresponding subdirectories. Subsequently, generated data associated with each PDB is stored consistently, providing better information control. While this actor is a small actor, it provides proper file management, a crucial component of CADD.

3.2 Ligand Parameterization

An MD simulation of a protein-ligand complex requires development of ligand force field parameters. Parameterization can be cumbersome and is commonly a multi-step process handled by a series of user scripts. To streamline this process, we developed a ligand parameterization composite actor (Figure 2), that follows the “gold standard” Amber protocol, using Antechamber [9] and Gaussian [10]. This actor has been parallelized using Kepler to distribute the workload and can therefore easily accept a large number of ligands as inputs. For each ligand, Antechamber assigns force field atom types, while Gaussian performs a minimization before calculating the electrostatic

potential (ESP), both at the HF/6-31G* level. Atomic partial charges are then assigned to reproduce the ESP using the RESP protocol [11]. The actor will read a directory of ligand PDB files and process them simultaneously. The PDB files will be moved into corresponding subdirectories to ensure that all output files are well organized along with their inputs when performing the calculations in parallel. These files and directories are then grouped together for the parameterization step as inputs, and lastly a distributor actor splits the task into smaller jobs that are executed in parallel. This composite actor subsequently outputs the required FRCMOD and PREPC files containing the force field parameters, which are reusable and easily shared.

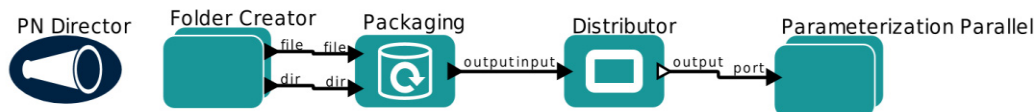


Figure 2: Kepler composite actor for the parameterization of small molecule ligands for MD

3.3 Receptor-ligand molecular dynamics simulations

The binding of a ligand to a receptor is a dynamic event. Small molecule compounds can assume many different binding poses, and receptor flexibility may change due to ligand binding. Therefore, it is important to consider the dynamic behavior of both ligands and receptors during CADD. One commonly applied method to describe the receptor-ligand dynamics is MD simulations of the complex. The steps to prepare an MD simulation can be routine but lengthy, especially when considering many different receptor-ligand complexes. To standardized and automate the process, we have developed a Kepler composite actor that simplifies the preparation of MD simulations of these complexes (see Figure 3). This actor takes the outputs generated from the ligand parameterization actor as the inputs. Furthermore, it requires a receptor PDB file in order to start the workflow. Once started, the job will run through three major components, described below, that collectively prepare and run an MD simulation of the user's system.



Figure 3: Layout of the Receptor-ligand molecular dynamic simulations actor.

Component I – Vina: Given PDB files of a ligand and a receptor, this module prepares the prerequisite files and docks the ligand into the receptor using Autodock Vina. The result is a PDB file that describes the “docked pose” of the ligand, or the conformation of the ligand when bound to the receptor.

Component II – PDB Modification: By concatenating the docked-pose PDB file to the PDB file of the receptor, component II first creates a merged ligand-receptor complex. Next, the receptor-ligand complex is assigned Amber force field parameters, and the topology and coordinate files required for MD are generated. Prior to simulating system dynamics, a restrained minimization is typically carried out to remove steric conflicts, which can cause MD programs to crash. In a final step, component II prepares the restraint files required during minimization.

Component III – Remote Login: This module of the composite actor prepares configuration files for MD simulation with NAMD [12] and writes submission scripts for running minimization, equilibration and production jobs on the XSEDE resource Stampede, located at the Texas Advanced Computing Center (see Figure 4). Future developments will enable users to employ alternate HPC resources. In order to take advantage of parallel computing, the files required for MD simulation that were generated in earlier steps must be moved to the HPC platform. Component III performs this operation, moving the prerequisite files to a user specified directory on a remote HPC resource. Once the files are transferred, component III executes and monitors minimization jobs on the HPC resource, generates the files necessary for a restrained MD equilibration, performs the restrained MD equilibration, and finally executes the production MD simulation.

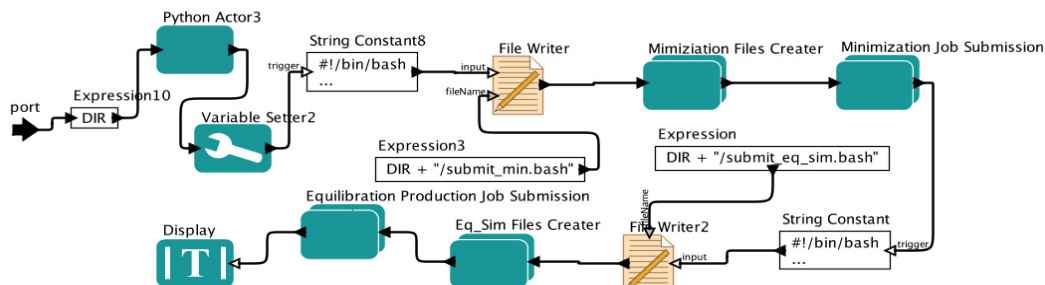


Figure 4: Breakdown of the remote login composite actor of the receptor-ligand dynamic simulation actor.

3.4 Receptor structural clustering

An MD simulation yields a “trajectory,” or a set of coordinates that represent the conformational states of the protein with or without a bound ligand as it evolves through time typically starting from an experimentally determined structure. With modern HPC resources, these trajectories can consist of thousands or even millions of conformations, which translates into giga- or terabytes of data, making structural analysis challenging. Fortunately, meaningful dataset reduction methods have been devised that extract representative conformations, or structures. These structures, are generally different than the experimentally determined structure(s), and the active sites display alternative conformations referred to as cryptic binding pockets [13-15], and can be exploited in subsequent VS.



Figure 5: Gromos receptor structural clustering actor.

Considering the size of contemporary MD datasets, an effective, integrated platform for studying protein dynamics will require workflow actors that leverage data reduction software in a single, cohesive, user-friendly framework. To that end, we developed a modular set of actors that process MD trajectories by GROMOS cluster analysis [16], a method that categorizes protein conformations based on structural similarity (see Figure 5). In the first processing step, the Trajectories Listing composite actor utilizes cpptraj, implemented in AmberTools, to concatenate short discontinuous trajectories into one long continuous trajectory. The PDB Creator and PDB Modifier actors then convert the input trajectory file to the PDB format required by Gromacs [17], and also strips solvent molecules and correct for periodic boundary conditions. The Atom Selection actor picks out the active site residues a user has predefined and creates a PDB file containing all the atom indices of the selected residues. The

files are then sent to the public NBCR Opal server, which aligns each trajectory conformation to a common reference and clusters the data using the Gromacs. We are planning to extend this workflow to include options for alternative clustering strategies.

3.5 Receptor and ligand preparation for docking

Docking programs, such as the widely used AutoDock [18] and AutoDock Vina (Vina) [19], provide scientists an estimate of the free energy change that occurs when a ligand binds to a receptor. Both AutoDock and Vina require PDBQT files that describe the coordinates, atomic partial charges, and AutoDock atom types of the ligand and the receptor. To streamline the conversion procedure, we have developed an actor that converts a receptor PDB to PDBQT file, which can be used by both AutoDock and Vina (see Figure 6). The actor uses the publicly available NBCR Opal server to perform the conversion, while Kepler monitors job scheduling and returns the output PDBQT file to the user's local machine. In the future, this actor will be extended to convert PDB to PDBQT for the ligand files as well.

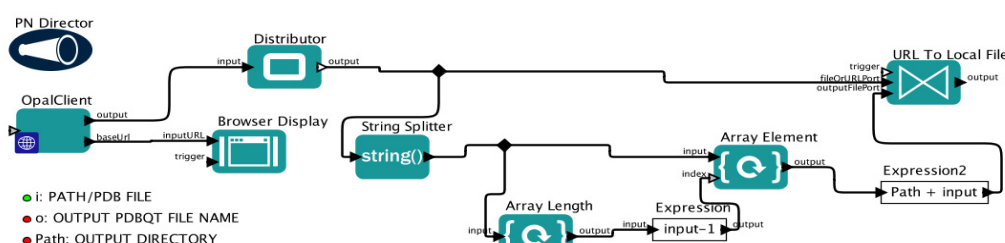


Figure 6: Receptor preparation for VS actor.

3.6 Ensemble based virtual screening

As previously stated, proteins are dynamic, and static crystal structures can offer a poor account of protein flexibility, particularly when it is pronounced. In a drug discovery context, this flexibility is manifested in the observance of so-called cryptic binding pockets [13-15], or ligand binding sites that are absent in a crystal structure but are present during an MD simulation. To incorporate these potential binding sites during VS, it is important to include an ensemble of protein receptor structures that models the flexibility of a receptor in solution. Here, we describe an actor that screens large ligand sets against different receptor conformations using Vina [19] (see Figure 7). Users supply a directory of receptor PDB files, a directory of ligand PDB files and grid information. Receptor PDB and ligand

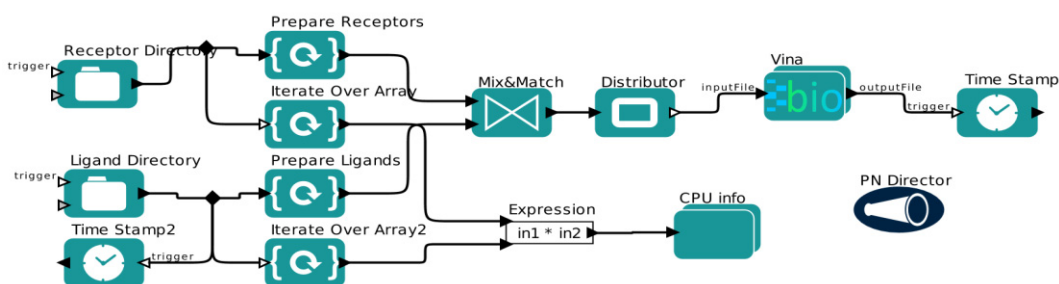


Figure 7: Virtual screening actor.

PDB files are converted to Vina specific PDBQT files. Every ligand is matched with each receptor once in the “Mix&Match” module, which organizes the large number of files generated in this protocol. The combinations are sent to Vina one by one for VS. Then, users have the option to either run VS locally or on the NBCR Opal server. This implementation has been made using the bioKepler extension [20]. Moreover, this actor carries out provenance and is able to output information regarding receptors, ligands, runtime and machines used for each run (data not shown).

3.7 Virtual screening performance statistics

During VS, small molecules are assigned a score *e.g.* during small molecule docking, the score is a predicted binding affinity of a small molecule to a receptor target, and those compounds predicted to bind more favorably receive a higher rank and are more likely to be experimentally assayed.

Performing VS using an ensemble of protein conformations may benefit the discovery effort, but it is also computationally demanding and scales linearly with the number of conformations. To improve computational efficiency, statistical methods can be used to select the ensemble that does the best job of separating known binders from the known non-binders in a small, experimentally characterized compound database. By carefully selecting the best performing ensemble, this protocol has the potential to reduce the computational expense of screening a much larger database of uncharacterized compounds.

We have developed an actor (see Figure 8) that incorporates the experimental status of a compound, *i.e.* binder or non-binder, the docking score of the compound in each receptor ensemble member, and returns the ensemble best able to discriminate known binders from known non-binders. Although there are various VS performance metrics available in the literature [21, 22], the area under the curve (AUC) of the Receiver Operating Characteristic (ROC) plot [23] is one of the most popular performance evaluation metrics and is used for our workflow. Part of the AUC's appeal is how easily it is interpreted. It represents the probability that a randomly selected binder will have a higher rank than a randomly selected non-binder [24, 25]. Consistent with this interpretation, an AUC value of 0.5 indicates the VS protocol performs randomly, while a value of 1 indicates the protocol ranks all of the binders ahead of all of the non-binders.

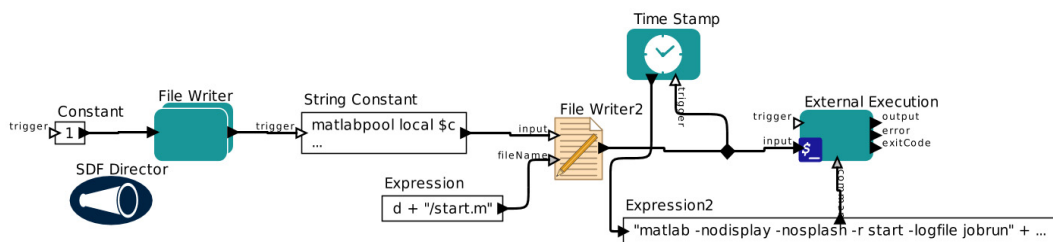


Figure 8: VS statistical performance actor utilizing Matlab

In practice, ensemble selection is complicated by the need to evaluate all possible combinations of receptor conformations, a combinatorial process described by the binomial coefficient. The workflow utilizes a series of Matlab [26] scripts to monitor performance of all possible ensembles of conformations. The scripts requires a comma-separated CSV file, containing ligand identification numbers, *e.g.* compound IDs in a database, a compound classifier, *i.e.* a 0 or 1, which labels non-binders and binders, respectively, and the remaining columns contain the docking scores for each receptor conformation. The scripts return the AUC value for all possible ensembles of receptor conformations, as well as the 95% confidence intervals, and p-values, which provide indications of the performance reliability and the statistical significance of the performance of each ensemble.

The calculations in Matlab are designed to utilize the Parallel Computing Toolbox in Matlab (parfor loops), although if the separate license required to use the toolbox is not available, the behavior

will default to standard loop iteration. The parallel option is highly recommended particularly for a large number of receptor structures, as these calculations otherwise become very time consuming.

4 Integrated web-services

The complexity of scientific applications needed in CADD often requires access to HPC resources. To ensure tasks are completed expediently, scalable and transparent support of distributed computing resources available on both HPC platforms and in the cloud is required of each Kepler workflow module. To meet this requirement, we use the Opal toolkit [27], which provides Scientific Software as a Service (SaaS) using standard and simple web interfaces. For example, scientific applications executed by the workflows are wrapped as SOAP-based web services that allow for programmatic and web-based application access, which is useful for a wide variety of applications. The programmatic capability allows transparent access of different workflow components, while the web-based service access provides a large number of NBCR applications to our affiliates and collaborators.

Using integrated web-services for scientific applications also aids our objective to develop a modular environment of interchangeable, customizable modules that can be used to create complex scientific workflows. As SaaS providers, we handle software installation configuration and upgrade transparently at the cyber-infrastructure level. With infrastructure complexities replaced by an easy-to-use interface, the full power of the modular workflow environment can be easily applied to pressing scientific problems.

The scientific applications, wrapped as Opal web services [28], can readily be deployed across distributed computing environments to accelerate completion of the scalable computations within the CADD framework. It is easy to access the scientific applications through the Opal web server, which provides a stable, reliable infrastructure for CADD and molecular simulations that can accommodate large throughput in an extensible, reproducible and reusable manner. This approach will allow flexible community resource sharing and, by providing the framework to incorporate ideas from a broad community of users, it will promote convergence toward a set of standardized best practices.

5 Workflow Dissemination

CADD workflows, in addition to other NBCR workflow products, are being made available through the NBCR website and GitHub [29]. We have enabled the NBCR workflows site to be searched and filtered easily through keywords describing the workflows' application, actors, program dependency, and other relevant terms, enabling the user to select the appropriate workflow for their needs. Upon selecting a desired workflow the user is taken to the workflow documentation and download options. The workflows will be distributed through GitHub to provide transparent version control. The user may either download the workflow itself, requiring a local installation of Kepler and dependent programs, or download the workflow as part of a Rock's Rolls [30] containing dependent programs. The Rock's Rolls facilitate the utilization of workflows in HPC environments.

6 Conclusions

We have developed a series of modular actors that can be integrated into a larger CADD framework, or be used as stand-alone tools. The modules described here have successfully been deployed on a number of different projects and are being optimized based on user feedback. These modules demonstrate the usability of Kepler scientific workflows in CADD with the aim to

standardize simulation and analysis, and to promote best practices within the molecular simulation and CADD communities. The workflows demonstrate usability in terms of file-management tasks, molecular simulation including ligand force field parameterization and management of job submission and monitoring on relevant HPC resources, as well as VS elements such as receptor structural clustering, docking and statistical analyses of the VS results. The modules developed here were partly intended for specific-use cases and will in the future be subdivided into smaller modules to avoid redundancy, and to make them amenable to incorporate into other framework uses.

Through our work, we have identified some novel Kepler capabilities that would be useful for the workflows in our science domain. A Kepler feature to allow for easier stitching together of several modules into a larger framework using a GUI framework would be helpful for more novice users. A particular function of this feature is for Kepler to check and match module output to subsequent module input requirements in the larger framework, and flag errors at the transitions, which can later be handled by the users in the design phase. The user can then quickly address any file format requirements, either by pulling in another module between the two causing the conflict, or manually provide/specify missing parameters. Furthermore, the current implementation for assembling several modules into a larger framework can be challenging when each module has its own PN or SDF director, as Kepler does not currently allow the PN actor to exist inside another PN or SDF director.

The current models are available for download on the NBCR website and have been integrated with NBCR web-services. Our lab is currently developing novel Kepler workflows designed for automation and standardizing of common tasks in CADD and molecular simulation. Additionally, we are developing domain specific interfaces for all NBCR workflows. These interfaces will integrate key visualization software, workflow modification, workflow execution management, and electronic lab book functions further optimizing the CADD process. We will solicit user feedback and use it to guide our efforts, to strengthening an ecosystem that encourages development and distribution of workflows with the simulation and CADD communities.

7 Acknowledgements

The authors would like to thank Leah Krause for fruitful discussions regarding the development of the workflows. This work is funded in part by a grant from the NVIDIA Foundation, an NIH New Innovator Award to REA OD-007237. Funding and support from the National Biomedical Computation Resource is provided through NIH P41 GM103426. bioKepler is funded by the National Science Foundation (NSF) DBI-1062565 under CI Reuse and Advances in Bioinformatics programs supported some of the IA, DC and JW. The Alfred Benzon foundation is thanked for postdoctoral funding to JS. This work used the Extreme Science and Engineering Discovery Environment (XSEDE) at San Diego Supercomputer Center (SDSC) and Texas Advanced Computing Center (TACC) using an allocation to REA with grant number TG-CHE060073N. XSEDE is supported by NSF grant number OCI-1053575.

References

1. Altintas, I., et al. *Kepler: an extensible system for design and execution of scientific workflows*. in *Scientific and Statistical Database Management, 2004. Proceedings. 16th International Conference on*. 2004.
2. McPhillips, T., et al., *Scientific workflow design for mere mortals*. *Future Generation Computer Systems*, 2009. **25**(5): p. 541-551.

3. Barseghian, D., et al., *Workflows and extensions to the Kepler scientific workflow system to support environmental sensor data access and analysis*. Ecological Informatics, 2010. **5**(1): p. 42-50.
4. Astakhov, V., et al., *Prototype of Kepler Processing Workflows For Microscopy And Neuroinformatics*. Procedia Computer Science, 2012. **9**(0): p. 1595-1603.
5. Moreau, L., et al., *Special Issue: The First Provenance Challenge*. Concurrency and Computation: Practice and Experience, 2008. **20**(5): p. 409-418.
6. Strickland, P.R., M.A. Hoyland, and B. McMahon, *Small-molecule crystal structure publication using CIF*, in *International Tables for Crystallography Volume G: Definition and exchange of crystallographic data*, S.R. Hall and B. McMahon, Editors. 2005, Springer Netherlands. p. 557-569.
7. Amaro, R.E., R. Baron, and J.A. McCammon, *An improved relaxed complex scheme for receptor flexibility in computer-aided drug design*. Journal of computer-aided molecular design, 2008. **22**(9): p. 693-705.
8. Murdock, S.E., et al., *Quality Assurance for Biomolecular Simulations*. Journal of Chemical Theory and Computation, 2006. **2**(6): p. 1477-1481.
9. Wang, J., et al., *Automatic atom type and bond type perception in molecular mechanical calculations*. Journal of Molecular Graphics & Modelling, 2006. **25**(2): p. 247-260.
10. Frisch, M.J., et al., *Gaussian 2009*, Gaussian, Inc.: Wallingford, CT, USA.
11. Bayly, C.I., et al., *A well-behaved electrostatic potential based method using charge restraints for deriving atomic charges: the RESP model*. J. Phys. Chem., 1993. **97**(40): p. 10269-10280.
12. Phillips, J.C., et al., *Scalable Molecular Dynamics with NAMD*. J. Comput. Chem., 2005. **26**: p. 1781-1802.
13. Amaro, R.E., et al., *Discovery of drug-like inhibitors of an essential RNA-editing ligase in Trypanosoma brucei*. Proceedings of the National Academy of Sciences of the United States of America, 2008. **105**(45): p. 17278-83.
14. Amaro, R.E., et al., *Remarkable loop flexibility in avian influenza N1 and its implications for antiviral drug design*. J Am Chem Soc, 2007. **129**(25): p. 7764-5.
15. Landon, M.R., et al., *Novel druggable hot spots in avian influenza neuraminidase H5N1 revealed by computational solvent mapping of a reduced and representative receptor ensemble*. Chem Biol Drug Des, 2008. **71**(2): p. 106-16.
16. Daura, X., W.F. van Gunsteren, and A.E. Mark, *Folding-unfolding thermodynamics of a beta-heptapeptide from equilibrium simulations*. Proteins, 1999. **34**(3): p. 269-80.
17. Pronk, S., et al., *GROMACS 4.5: a high-throughput and highly parallel open source molecular simulation toolkit*. Bioinformatics, 2013. **29**(7): p. 845-854.
18. Morris, G.M., et al., *AutoDock4 and AutoDockTools4: Automated docking with selective receptor flexibility*. Journal of Computational Chemistry, 2009. **30**(16): p. 2785-2791.
19. Trott, O. and A.J. Olson, *AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading*. J Comput Chem, 2010. **31**(2): p. 455-61.
20. Altintas, I., et al., *Challenges and approaches for distributed workflow-driven analysis of large-scale biological data: vision paper*, in *Proceedings of the 2012 Joint EDBT/ICDT Workshops2012*, ACM: Berlin, Germany. p. 73-78.
21. Truchon, J.F. and C.I. Bayly, *Evaluating virtual screening methods: good and bad metrics for the "early recognition" problem*. J Chem Inf Model, 2007. **47**(2): p. 488-508.
22. Zhao, W., et al., *A statistical framework to evaluate virtual screening*. BMC Bioinformatics, 2009. **10**: p. 225.
23. Fawcett, T., *An introduction to ROC analysis*. Pattern Recognition Letters, 2006. **27**(8): p. 861-874.

24. Nicholls, A., *What Do We Know?: Simple Statistical Techniques that Help*, in *Chemoinformatics and Computational Chemical Biology*, J. Bajorath, Editor. 2011, Humana Press. p. 531-581.
25. Jain, A.N., *Bias, reporting, and sharing: computational evaluations of docking methods*. *Journal of Computer-Aided Molecular Design*, 2008. **22**(3-4): p. 201-212.
26. *Matlab*, 2011, The MathWorks Inc.: Natick, Massachusetts.
27. Krishnan, S., et al. *Design and Evaluation of Opal2: A Toolkit for Scientific Software as a Service*. in *Services - I, 2009 World Conference on*. 2009.
28. Krishnan, S., et al. *Opal: SimpleWeb Services Wrappers for Scientific Applications*. in *Web Services, 2006. ICWS '06. International Conference on*. 2006.
29. *Github*, 2014: <https://github.com>.
30. Bruno, G., et al. *Rolls: modifying a standard system installer to support user-customizable cluster frontend appliances*. in *Cluster Computing, 2004 IEEE International Conference on*. 2004.